Providing End-to-End User Authentication for Host Access Using Digital Certificates

BACKGROUND OF THE INVENTION

Field of the Invention

The present invention relates to a computer system, and deals more particularly with a method, system, and computer program product for providing end-to-end user authentication using digital certificates for accessing host applications and data without requiring modification of existing host applications.

Description of the Related Art

One of the challenges facing information services ("IS") professionals today is the difficulty of providing secure access to legacy mainframe host data and applications from modern personal computer-based ("PC-based") applications. As more large companies move to provide business integration and self-service on the World Wide Web (hereinafter, "Web), there is most often data that is crucial to this movement, but which is based on (and is only accessible through) legacy mainframe host applications. These host applications and their data have, from their origin, been typically protected through the use of the program product commonly referred to as "RACF" (Resource Access Control Facility) or other similar mainframe-based security systems. ("RACF" is a registered trademark of the IBM Corporation.) These mainframe-based security

systems typically require a user identification and password in order to gain access to the protected applications and data. Therefore, when a user tries to access data or legacy applications on a host mainframe from a client workstation over a network connection, they normally must provide a separate user identification and password to the host application to satisfy the security requirements of the host security system in addition to the user identification and password they use to access the modern environments (e.g. to access the Internet or Web). This double entry of identifying information is not only redundant but tedious for the user as well.

With the wide-spread use of SSL (Secure Sockets Layer) and certifiable digital certificates for providing security in today's PC-based computing environments, there is a desire to use a client certificate as the basis for a "single system log on" to all of a user's Internet-based applications. This includes applications that provide access to legacy host applications and/or data such as IBM's Host-On-Demand, Personal Communications, and Host Publisher products. Digital certificates are used to authenticate entities, as is well known in the art. U. S. Patent _____ (serial number 09/064,632, filed 12/10/98), which is titled " Certificate Based Security in SNA Data Flows", teaches a technique whereby digital certificates are transported in appropriate Systems Network Architecture ("SNA") data flows between a client and a host for identifying the user to the host application, but this existing technique requires those host programs which authenticate the user to RACF (or other host access control facility) to be modified to use the certificate instead of the traditional userid and password. This requires an enterprise to upgrade each of its application subsystems in order to achieve the benefits. So for some enterprises, the previous approach may be impractical and unacceptable.

Accordingly, what is needed is a technique that provides a single system log on that allows a host-based, legacy security system to authenticate a client from the newer PC-based, distributed computing environments without requiring the client to supply an additional user name (or other user identifier) and password. This technique must allow current legacy applications to function without requiring any modification thereof.

SUMMARY OF THE INVENTION

An object of the present invention is to provide a technique for using a single system log on to access legacy host applications and data in a distributed networking environment.

Another object of the present invention is to provide this technique without requiring any modification to the existing legacy host applications.

Yet another object of the present invention is to provide this technique by using digital certificates to authenticate clients.

Other objects and advantages of the present invention will be set forth in part in the description and in the drawings which follow and, in part, will be obvious from the description or may be learned by practice of the invention.

To achieve the foregoing objects, and in accordance with the purpose of the invention as broadly described herein, the present invention provides a computer program product, system,

and method for providing end-to-end user authentication for legacy host application access in a computing environment. This technique comprises: establishing a secure session from a client machine to a server machine using a digital certificate representing the client machine or a user thereof; storing the digital certificate at the server machine; establishing a session from the server machine to a host system using a legacy host communication protocol; passing the stored digital certificate from the server machine to a host access security system; using, by the host access security system, the passed digital certificate to locate access credentials for the user; accessing a stored password or a generated password substitute representing the located credentials; and using the stored password or the generated password substitute to transparently log the user on to a secure legacy host application executing at the host system.

The digital certificate may be an X.509 certificate. The communication protocol may be a 3270 emulation protocol. Or, it may be a 5250 emulation protocol or a Virtual Terminal protocol. The host access security system may be a Resource Access Control Facility (RACF) system.

The technique may further comprise: requesting by the legacy host application, responsive to establishing the session, log on information for the user; responding to the request for log on information by sending a log on message with placeholders from the client machine to the server machine, these placeholders representing a user identification and a password of the user; and substituting a user identifier associated with the located access credentials and the stored password or the generated passticket for the placeholders in the log on message.

In one aspect, the server machine may be a Web application server machine. In this case, the technique may further comprise: requesting by the legacy host application, responsive to establishing the session, log on information for the user; and responding to the request for log on information by supplying a user identifier associated with the located access credentials and the stored password or the generated passticket at the server machine.

The present invention will now be described with reference to the following drawings, in which like reference numbers denote the same element throughout.

BRIEF DESCRIPTION OF THE DRAWINGS

Figure 1 is a block diagram of a computer workstation environment in which the present invention may be practiced;

Figure 2 is a diagram of a networked computing environment in which the present invention may be practiced;

Figure 3 illustrates message flows for authentication of a user between a typical PC-based user and a host-based application, according to the prior art;

Figure 4 illustrates message flows for authentication of a user according to a first preferred embodiment of the present invention operating in a distributed computing environment;

Figure 5 illustrates message flows for authentication of a user according to a second preferred embodiment of the present invention operating in a Web Application environment; and

Figure 6 depicts the format of an X.509 certificate that may be used with the preferred embodiments of the present invention.

DESCRIPTION OF THE PREFERRED EMBODIMENTS

Figure 1 illustrates a representative workstation hardware environment in which the present invention may be practiced. The environment of Fig. 1 comprises a representative single user computer workstation 10, such as a personal computer, including related peripheral devices. The workstation 10 includes a microprocessor 12 and a bus 14 employed to connect and enable communication between the microprocessor 12 and the components of the workstation 10 in accordance with known techniques. The workstation 10 typically includes a user interface adapter 16, which connects the microprocessor 12 via the bus 14 to one or more interface devices, such as a keyboard 18, mouse 20, and/or other interface devices 22, which can be any user interface device, such as a touch sensitive screen, digitized entry pad, etc. The bus 14 also connects a display device 24, such as an LCD screen or monitor, to the microprocessor 12 via a display adapter 26. The bus 14 also connects the microprocessor 12 to memory 28 and long-term storage 30 which can include a hard drive, diskette drive, tape drive, etc.

The workstation 10 may communicate via a communications channel 32 with other computers or networks of computers. The workstation 10 may be associated with such other

computers in a local area network (LAN) or a wide area network, the workstation 10 can be a client in a client/server arrangement with another computer, etc. All of these configurations, as well as the appropriate communications hardware and software, are known in the art.

Figure 2 illustrates a data processing network 40 in which the present invention may be practiced. The data processing network 40 may include a plurality of individual networks, such as wireless network 42 and network 44, each of which may include a plurality of individual workstations 10. Additionally, as those skilled in the art will appreciate, one or more LANs may be included (not shown), where a LAN may comprise a plurality of intelligent workstations coupled to a host processor.

Still referring to Figure 2, the networks 42 and 44 may also include mainframe computers or servers, such as a gateway computer 46 or application server 47 (which may access a data repository 48). A gateway computer 46 serves as a point of entry into each network 44. The gateway 46 may be preferably coupled to another network 42 by means of a communications link 50a. The gateway 46 may also be directly coupled to one or more workstations 10 using a communications link 50b, 50c. The gateway computer 46 may be implemented utilizing an Enterprise Systems Architecture/370 available from IBM, an Enterprise Systems Architecture/390 computer, etc. Depending on the application, a midrange computer, such as an Application System/400 (also known as an AS/400) may be employed. ("Enterprise Systems Architecture/370" is a trademark of IBM; "Enterprise Systems Architecture/390", "Application System/400", and "AS/400" are registered trademarks of IBM.)

The gateway computer 46 may also be coupled 49 to a storage device (such as data repository 48). Further, the gateway 46 may be directly or indirectly coupled to one or more workstations 10.

Those skilled in the art will appreciate that the gateway computer 46 may be located a great geographic distance from the network 42, and similarly, the workstations 10 may be located a substantial distance from the networks 42 and 44. For example, the network 42 may be located in California, while the gateway 46 may be located in Texas, and one or more of the workstations 10 may be located in New York. The workstations 10 may connect to the wireless network 42 using a networking protocol such as the Transmission Control Protocol/Internet Protocol ("TCP/IP") over a number of alternative connection media, such as cellular phone, radio frequency networks, satellite networks, etc. The wireless network 42 preferably connects to the gateway 46 using a network connection 50a such as TCP or UDP (User Datagram Protocol) over IP, X.25, Frame Relay, ISDN (Integrated Services Digital Network), PSTN (Public Switched Telephone Network), etc. The workstations 10 may alternatively connect directly to the gateway 46 using dial connections 50b or 50c. Further, the wireless network 42 and network 44 may connect to one or more other networks (not shown), in an analogous manner to that depicted in Fig. 2.

Software programming code which embodies the present invention is typically accessed by the microprocessor 12 of workstation 10 and server 46 from long-term storage media 30 of some type, such as a CD-ROM drive or hard drive. The software programming code may be embodied

on any of a variety of known media for use with a data processing system, such as a diskette, hard drive, or CD-ROM. The code may be distributed on such media, or may be distributed to users from the memory or storage of one computer system over a network of some type to other computer systems for use by users of such other systems. Alternatively, the programming code may be embodied in the memory 28, and accessed by the microprocessor 12 using the bus 14. The techniques and methods for embodying software programming code in memory, on physical media, and/or distributing software code via networks are well known and will not be further discussed herein.

A user of the present invention may connect his computer to a server using a wireline connection, or a wireless connection. Wireline connections are those that use physical media such as cables and telephone lines, whereas wireless connections use media such as satellite links, radio frequency waves, and infrared waves. Many connection techniques can be used with these various media, such as: using the computer's modem to establish a connection over a telephone line; using a LAN card such as Token Ring or Ethernet; using a cellular modem to establish a wireless connection; etc. The user's computer may be any type of computer processor, including laptop, handheld or mobile computers; vehicle-mounted devices; desktop computers; mainframe computers; etc., having processing and communication capabilities. The remote server and the intermediary, similarly, can be one of any number of different types of computer which have processing and communication capabilities. These techniques are well known in the art, and the hardware devices and software which enable their use are readily available. Hereinafter, the user's computer will be referred to equivalently as a "workstation", "device", or "computer", and

use of any of these terms or the term "server" refers to any of the types of computing devices described above.

In the preferred embodiments, the present invention is implemented as one or more modules (also known as "objects" in object-oriented programming languages) of one or more computer software programs. This computer software will be used in an environment where a user in a modern distributed computing environment is accessing a host legacy application where the application and/or data it uses is protected by a host-based security system (such as RACF).

The preferred embodiments of the present invention enable a user to provide a single system log on for accessing applications and data during the user's session, whether the applications and data are available from a modern PC-based environment such as the Internet or whether they are available only through a legacy host application which is protected by a host-based security system.

The preferred embodiments of the present invention will now be discussed with reference to Figs. 3 through 6.

Fig. 3 illustrates message flows that may be used for authenticating a user by a legacy host application according to the prior art. When a user at a client device wishes to work with a legacy host application and/or data, the client device must use some form of emulation or emulator product to allow communication between the distributed computing environment and

the host application. In the example of Fig. 3, the client is using an emulator product which uses the TN3270 emulation protocol. The TN3270 protocol is used to provide emulation of the "3270 data stream", as is well known to those familiar with the art. The 3270 data stream is frequently used for information transfer to and from legacy host applications, and is so named because it was originally designed for use with IBM Model 3270 client workstations.

Note that while the examples describing the present invention are discussed with reference to the 3270 data stream format, this is for purposes of illustration and not of limitation. Other data stream formats may be used alternatively. Another commonly used data stream format for communicating with legacy host applications is referred to as a "5250 data stream", originally designed for communicating with IBM Model 5250 workstations. The TN5250 emulation protocol is used with a 5250 data stream. Yet another commonly used data stream is an ASCII data stream, commonly referred to as a "Virtual Terminal" or "VT" data stream.

When a user in the modern distributed computing environment begins working in a secure environment, he is asked to provide a user identification and password which is used to authenticate who the person is and typically what resources this particular user is authorized to access.

To begin the process depicted in Fig. 3, the client, using software such as emulator client 300, negotiates at 325 with the server 305 for the services required to allow the user to communicate with the host application. At 330, the server 305 opens an SNA session with the

host system 310 on behalf of the client 300. At 335, the host application sends application data formatted as a 3270 data stream to the client 300. This data passes through the server 305 where it is transformed from a 3270 data stream to a data stream (such as a standard TN3270 data stream) suitable for the emulator client. This initial data from the host application is typically some type of "log on" screen asking for a user name (or other user identifier) and password. It should be noted that at this point, no interaction with the RACF program 315 has occurred, since the host application is responsible for providing the client's user name and password to the RACF system for authentication. At 340, the client signifies a log on to the host application by providing a user name and password. This data flows to the server 305 where it is transformed to a 3270 data stream format, which is then sent to the host system 310 (and application) for processing. The host application then forwards (at 345) the user name and password to the RACF system 315 for authentication. The authentication is performed using the supplied user name and password. The RACF system then responds 350 to the host application with either a success or failure of the authentication process. If the authentication was successful, the host application then begins normal communication with the client (as shown at 360).

As the move towards distributed computing and use of the Internet continues, it is anticipated that enterprises will require use of digital certificates and standardized security protocols (such as SSL) for authentication of users who wish to access the enterprise applications and data. A digital certificate may be generated for a user using techniques which are known in the art, for example by contacting a certificate authority which issues such certificates. Techniques for obtaining a digital certificate do not form part of the present invention. Once a digital

certificate is generated for a user, the certificate may be used according to the present invention for accessing resources in the modern distributed computing environment which are protected by host access control facilities such as RACF. A certificate conforming to the X.509 standard (hereinafter referred to as an "X.509 certificate") is used in the preferred embodiments of the present invention, although other digital certificate formats may be used alternatively without deviating from the scope of the present invention. "X.509" is an International Telecommunication Union (ITU) Recommendation and International Standard that defines a framework for providing authentication. (See ITU Recommendation X.509 (1997), titled "Information Technology - Open Systems Interconnection - The Directory: Authentication Framework", dated 8/97. This information is also published in International Standard ISO/IEC 9594-8 (1995).) A certificate format is defined in this standard. Certificates created according to this international standard, in the defined format, are referred to as "X.509 certificates".

The format of an X.509 certificate is shown in Fig. 6. Hereinafter, references to "certificates" refer to the type of information shown in Fig. 6. The subject field 660 identifies the entity (e.g. the user) to which this certificate was issued. The preferred embodiments of the present invention use this subject field to identify the user, where this identification process is automatic and transparent to the user himself. In this manner, the need for the user to explicitly (and redundantly) re-identify himself for purposes of communicating with a legacy host application and its security system is avoided.

The technique with which a digital certificate is used for authentication is well known in

the art and will not be described in detail herein. For purposes of the example embodiments discussed below with reference to Figs. 4 and 5, it is assumed that the user has already been issued a digital certificate and that this certificate is stored in such a manner that it is locally accessible to the client software operating on the user's workstation.

Fig. 4 illustrates message flows in a distributed computing environment for authenticating a user according to a first preferred embodiment of the present invention. As indicated in Fig. 4, the emulator client at 420 initiates an SSL session and provides a digital certificate to the server 405. As stated above, it is assumed that this digital certificate is already available at the client machine.

The server authenticates the client using the client's digital certificate as part of the existing SSL session establishment procedure. The server then, according to the present invention, caches (or otherwise stores) the certificate for later use. Negotiation 425 of the session parameters between the client and the server then occurs as in the prior art. Since this client 400 desires to interact with the host system 410, the server 405 initiates 430 an SNA session on behalf of the client with the host system 410. The host then responds at 435 with the host application initiation data (e.g. a request for supplying user identification and password information), which flows through the server 405 to the client 400. The client software at 440 responds to the server 405 with an indication that the client wishes to log on to the host application. Rather than prompting the user to explicitly identify himself (for example, by typing in his identifier and password) as in the prior art (see the prior discussion of element 340 of Fig. 3), the present

invention automatically (and transparently to the user) inserts placeholders in response 440. Note that the placeholder is represented in 440 as having the syntax "$$user$$". This is merely an example of a placeholder syntax which may be used. Alternative placeholder syntax may be substituted without deviating from the inventive concepts disclosed herein. Furthermore, separate placeholders may be used for the user identifier and the password, or a single placeholder may be used for both. What is required is that the client software 400 and the software operating at server 405 agree on a particular syntax to represent that a placeholder is being transmitted.

Upon receiving message 440 and detecting the presence of the placeholders, the server locates the client's cached X.509 certificate, which was obtained at 420 during SSL session establishment. This cached certificate is then passed in flow 445 to the host-based RACF 415 software. The RACF system extracts the user's identification from the subject field of the client certificate, and uses this information to locate the user's stored credentials and access privileges. For example, the value of the subject field may be used as a key to access a stored repository of credentials, where the data in the repository has been previously created by a person such as a systems administrator. Or, the subject field may be used to access a lookup table of such information, or to access a lookup table which provides a correlation to a key used to access a credential repository (such as a subject value-to-credential key correlation). The manner in which the credentials are stored is outside the scope of the present invention.

The RACF secured sign-on procedure is then invoked at the RACF implementation 415, using techniques which are known in the art. "RACF secured sign-on" is a procedure for enabling

clients to sign on to a host and communicate securely without sending RACF passwords across a network. Instead, a dynamically-generated short-lived credential referred to as a "passticket" is generated by the RACF software as a password substitute. Passtickets, and the procedure with which they are generated, are known in the art. As an alternative (for example, in other host access systems other than RACF) to generating a passticket, an actual password may be retrieved by the host access security system, where this password may then be used directly instead of using a passticket as a password substitute. Hereinafter, references to "passticket" are to be interpreted as referring equivalently to use of a password supplied by the host access security system.

According to the present invention, the passticket represents the access privileges for the user identified by the subject field of the digital certificate transmitted at 445. The RACF software 415 sends 450 this passticket to the server, along with the user name (or identification) to which it corresponds (i.e. the user name associated with the credentials represented by the passticket). The server at 455 then inserts the returned user name and passticket into the 3270 data stream in place of the placeholders (completing the log on request message 440 from the client software 400), and sends the resulting data stream to the host 410. Using this passticket and user identification data, the legacy host application can determine the user's access privileges in the manner with which it has already been programmed. The host application and the client interact as shown at 460, without requiring modification of the host application.

Fig. 5 illustrates message flows of the present invention in a Web application environment according to a second preferred embodiment of the present invention. As indicated in Fig. 5, the

client's browser 500 at 520 initiates an SSL session and provides a digital certificate (preferably an X.509 certificate, as previously described in the discussion of the first preferred embodiment with reference to the certificate 600 of Fig. 6) to the Web application server 505. The Web application server 505 authenticates the client using this digital certificate as in the prior art and, according to the present invention, caches (or otherwise stores) the certificate for later use. The client 500 then begins interaction with the Web application server 505 at 525. Since the client desires to interact with the host system 510, the Web application server 505 initiates a 3270 session (using the TN3270 emulation protocol) at 530 with the host server 510 located on a host machine. (As described above, other data stream formats and other emulation procedures may be used alternatively without deviating from the inventive concepts disclosed herein.) The 3270 data stream application data flows at 535 from the host application at 510 to the Web application server 505. (Note that in the scenario depicted in Fig. 5, the Web application server 505 is functioning as a proxy for client 500, intercepting and responding to messages on behalf of the client software. Thus, the client software 500 may operate without change in this preferred embodiment.)

The Web application server at 540 signifies to the host server that the client wishes to log on to the application. Typically, a callout procedure exists in the application software executing on the Web application server 505 which, in the prior art, would prompt the user with special input prompts (for example, by presenting a Web page form for the user to fill in) to supply his identification and password information. According to the present invention, however, the Web application server application software automatically and transparently uses the cached client's

certificate 600 (which was obtained during flow 520) to supply this information. The Web application server locates the previously-stored client certificate, and includes it in message 540 which is sent to the RACF implementation 515. As described above with reference to Fig. 4, RACF uses the digital certificate to extract a user name (from subject field 660) and to generate a passticket representing the credentials of that user after accessing stored credential information. At 545, RACF returns the user name and passticket through the host server 510 to the Web application server 505. According to the present invention, in response to receiving message 545, the Web application server 505 inserts the user name and passticket information into the 3270 data stream expected by the host application and sends it, at 550, to the host server 510. The legacy host application uses this information to allow the user to access protected applications and/or data, with requiring changes to the host application itself. Traffic flows between the host server 510 and the Web application server 505 at 555 as in the prior art. Interaction between the legacy host application and client 500 (shown at 560) occurs also as in the prior art.

While the preferred embodiment of the present invention has been described, additional variations and modifications in that embodiment may occur to those skilled in the art once they learn of the basic inventive concepts. In particular, alternative data streams (such as a 5250 data stream or a VT data stream) may be used which provide the communications between the user's modern PC-based computer system and the legacy host applications and data. Further, security software other than the IBM RACF software may be used for protecting host-based assets. Therefore, it is intended that the appended claims shall be construed to include both the preferred embodiment and all such variations and modifications as fall within the spirit and scope of the

invention.